```
// E – empty (nič tam už nie je)
<start>              ->      <func> <scope>

// Hlavne telo programu
<scope>       ->      SCOPE EOL <scope_body> END SCOPE

<scope_body>->        E
<scope_body>->        DIM ID AS <type> <dim_n> <scope_body>
<scope_body>->        <body> <scope_body>

<cond_body>->         E
<cond_body>->         <body><cond_body>

<body>        ->      E
<body>        ->      EOL
<body>        ->      ID = <expr> EOL
<body>        ->      <condition>
<body>        ->      <loop>
<body>        ->      PRINT <printable>
<body>        ->      INPUT ID EOL

<dim_n>       ->      EOL
<dim_n>       ->      = <expr> EOL

<printable>   ->      <expr> ; <printable_n>

<printable_n> ->      <printable>
<printable_n> ->      EOL

<expr>        ->      val <operation>
<expr>        ->      ID <operation>
<expr>        ->      ID(<param>)

<b_expr>      ->      <expr> <b_op> <expr>
<b_expr>      ->      <expr>

<condition>   ->      IF <b_expr> THEN EOL <cond_body> ELSE EOL <cond_body> END IF EOL

<loop>        ->      DO WHILE <b_expr> EOL <cond_body> LOOP EOL

<param>       ->      E
<param>       ->      ID <params>

<params>      ->      E
<params>      ->      , ID <params>




// Deklaracia a definicia funkcii
<func>        ->      E
<func>        ->      <declare>
<func>        ->      <define>

<declare>     ->      DECLARE FUNCTION ID (<func_param>) AS <type> EOL <func>

<define>      ->      FUNCTION ID (<func_param>) AS <type> EOL <func_dim> END FUNCTION EOL<func>

<func_param>->        <first_func_p>
<func_param>->        E
```

```
<first_func_p>->      ID AS <type> <func_params>

<func_params>->       , ID AS <type> <func_params>
<func_params>->       E

<func_dim>    ->      E
<func_dim>    ->      DIM ID AS <type> <dim_n> <func_dim>
<func_dim>    ->      <func_body> <func_dim>


<func_body>  ->       E
<func_body>  ->       EOL
<func_body>  ->       ID = <expr> EOL
<func_body>  ->       <func_condition>
<func_body>  ->       <loop>
<func_body>  ->       PRINT <printable>
<func_body>  ->       INPUT ID EOL
<func_body>  ->       RETURN <expr> EOL

<f_cond_body>->       E
<f_cond_body>->       <func_body><f_cond_body>

<func_condition>->    IF <b_expr> THEN EOL <f_cond_body> ELSE EOL <f_cond_body> END IF EOL


<type>        ->      INTEGER
<type>        ->      STRING
<type>        ->      DOUBLE

<operation>   ->      <op> <expr>
<operation>   ->      E

<op>          ->      +
<op>          ->      -
<op>          ->      *
<op>          ->      /
<op>          ->      \

<b_op>        ->      <
<b_op>        ->      >
<b_op>        ->      <=
<b_op>        ->      >=
<b_op>        ->      =
<b_op>        ->      <>
```